AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (currently amended) A method of operating an authenticating server system for authenticating ~~users at client terminals connected~~a user of a client application provided on a client terminal via a data communications network, the server system being arranged to control access to a plurality of documents stored on a resource server~~,~~ connected to said data communications network, said method comprising performing the following in said server system:

storing in the resource server authentication details including a unique identifier for the client application of the user and access status data of authorized users;

~~receiving~~ validating the unique identifier at the resource server using authentication data ~~for a user from a~~received from the user's client terminal ~~of the user~~ and ~~validating at the resource server said authentication data~~ by reference to said stored user authentication details;

storing at the resource server:

(1) ~~an~~ the validated identifier for the client ~~terminal~~application, the identifier indicating ~~said terminal to be~~ the client application is associated with a currently authenticated ~~terminal~~user; and

(2) the access status of the user of the ~~currently authenticated terminal~~client application providing the validated identifier; and

enabling said resource server to validate a request for said document sent by the user from the client ~~terminal of said user, which request includes said~~ application, the request including a validated identifier, by checking that said stored access status includes said document.

2. (currently amended) A method according to claim 1, wherein said identifier is transmitted in a cookie to said client ~~terminal~~application.

3. (currently amended) A method according to claim 1, wherein said identifier is received from said client ~~terminal~~ application with said authentication data.

4. (currently amended) A method according to claim 3, wherein a new identifier is issued to said client ~~terminal~~ application if said authentication data is invalid.

5. (currently amended) A method according to claim 4, wherein said identifier comprises data indicating the number of times an invalid authenticator has been received from said client ~~terminal~~application.

6. (currently amended) A method according to claim 5, wherein said method comprises issuing no further identifier to said client ~~terminal~~ application if an identifier received from said client ~~terminal~~ application indicates that a predetermined number of invalid authenticators have been received from said client ~~terminal~~application.

7. (currently amended) A method according to claim 1, comprising timing out said identifier as an identifier of ~~a terminal~~ an application of a currently authenticated user if no document request is received from said client ~~terminal~~ application for a predetermined period.

8. (currently amended) A method according to claim 1, comprising authenticating said user for access to a plurality of Web servers located in the same Internet domain; and

enabling each of said Web servers to validate document requests from the client ~~terminal~~application, which requests include said identifier, by checking said status data on receipt of a document request.

9. (currently amended) A method of operating an authenticating server system for authenticating users at client terminals remotely connected via a data communications network, the server system being arranged to control access to a plurality of documents stored on a plurality of resource servers connected to said data communications network, said method comprising performing the following steps in said server system:

storing in at least one of the resource servers authentication details including a unique identifier for each client application via which a user seeks access to one or more of said plurality of documents and access status data of authorized users;

~~performing at the~~ validating for each said user at said at least one of the resource servers ~~remote~~ the unique identifier at the resource server using authentication ~~of a user~~ data received from the user's client terminal and by reference to said stored user authentication details and during said remote authentication step generating ~~the~~ said access status data of the user, distinguishing said user from other users which are not currently authenticated, and generating a secret encryption key to be shared with said user;

storing at said at least one resource server:

~~storing said access status data in the at least one of the resource servers to check an authentication status of said user by using an~~

(1) the validated identifier for the client application, the identifier indicating the client application is associated with a currently authenticated user; and

(2) the access status data of the user by using the validated identifier for the client application ~~terminal~~ received in a service request to check the stored access status data; and

storing said shared secret key in a data store accessible by at least one of said resource servers for use during communications with said user.

10. (currently amended) A method according to claim 9, wherein said remote authenticating step comprises issuing a challenge to the client ~~terminal~~application, receiving a response to said challenge, and verifying said response.

11. (previously presented) A method according to claim 9, further comprising updating said access status data for an authenticated user following said storing step.

12. (previously presented) A method according to claim 11, wherein said updating step is performed in response to a time-out associated with said access status data.

13. (previously presented) A method according to claim 11, wherein said updating step is performed in response to access by one of said resource servers to said access status data.

14. (currently amended) A method according to claim 12, wherein said updating step is performed in response to a request by the client ~~terminal~~application.

15. (canceled)

16. (currently amended) A method according to claim 9, wherein said authentication step comprises issuing said identifier to the client ~~terminal~~application.

17. (previously presented) A method according to claim 9, wherein said access status data is stored in a data store of at least one of said resource servers.

18. (previously presented) A method according to claim 9, wherein said authentication details include data identifying the rights of access of individual users to one or more of said resource servers.

19. (previously presented) An authenticating server system adapted to perform the method of claim 1.

---

20. (new) A method of logging on a user connected to a data communications network via a client terminal to an authenticating server system, the authentication server system being arranged to control access to a document requested by the user and stored on a resource server connected to the data communications network, the method of logging the user on comprising the steps of:

the user sending a request for access to the document to the resource server using an application client provided on the client terminal;

receiving the request for access at the resource server and determining if the request contains a unique identifier for the user, and if not performing the following:

the resource server generating a unique identifier for the client application which enables the user to be uniquely identified in subsequent requests sent by the client application to the resource server;

the resource server storing the unique identifier as an unvalidated unique identifier and sending the identifier to the user;

receiving the identifier at the application client terminal and associating the identifier for the client application with authentication data for the user of the client application in one or more subsequent requests sent to the resource server; and

upon receiving said one or more subsequent requests at the resource server, authenticating the user by using:

the unique identifier to identify the client application of the user; and

the associated authentication data to validate the unique identifier.

21. (new) An authentication server system arranged to perform steps in a method of logging on a user to the authentication server system as claimed in claim 20, the server system being arranged to perform the steps of:

receiving the request for access from the client application of the user;

determining if the request contains a unique identifier for the client application, and if not:

generating a unique identifier for the client application which enables the client application to be uniquely identified in subsequent requests sent by the user of the client application to the resource server;

storing the unique identifier as an unvalidated unique identifier at the resource server;

sending the unvalidated identifier to the user;

receiving a subsequent request containing an unvalidated identifier together with authentication data enabling the user to be authenticated; and

using the unique identifier and associated user authentication data to validate the unique identifier and authenticate the user.

22. (new) A client application provided on a user's client terminal and arranged to perform steps in a method of logging on the user to the authentication server system as claimed in claim 1, the client terminal being arranged to perform the steps of:

sending a request for access to the document to the resource server;

receiving the identifier provided by the authentication server system;

generating authentication data for the user;

associating the authentication data with the identifier; and

returning the identifier with the associated authentication data in one or more subsequent requests for access to the resource server.